International Journal of Scientific Research in Computer Science, Engineering and Information Technology
© 2018 IJSRCSEIT | Volume 3 | Issue 5 | ISSN : 2456-3307

Study on Task Scheduling and Resource Allocation in Cloud Computing Using ACO

S.Krishnaprasad[1], Dr. P. Srivaramangai[2]

[1]Ph. D. Research Scholar, Department of Computer Science, Maruthu pandiyar College of Arts & Science, Thanjavur, Tamilnadu, India

[2]Associate Professor, Department of Computer Science, Maruthu pandiyar College of Arts & Science , Thanjavur. Tamilnadu, India

ABSTRACT

The development of cloud computing infrastructures carries innovative ideas to make prove and control computing system by means that of the flexibility present with virtualization technologies. In this framework, it focuses on two important goals. Initial to afford virtualization and cloud computing infrastructures to make distributed large scale computing platforms from completely different cloud providers approved to run software involving large volumes of computation power. Subsequently developing methods to invent these infrastructures are more dynamic. This method provides inter cloud live migration planning and innovative ideas to utilize the inherent dynamic environment of distributed clouds. A load balancing process is considered as an important optimization process for utilizing dynamic resource allocation in cloud computing. In order to achieve maximum resource efficiency and extensibility in a quick manner and this process is concerned with multiple objectives for an efficient distribution of loads among virtual machines. During this realm, analyze new algorithms, as well as development of novel algorithms, is highly desired for technological improvement and long-term progress in resource allocation application in cloud computing. In this paper, a cloud load reconciliation policy ant Colony improvement (ACO) inspired by ant Systems is introduced. Consequently, this paper provides an general idea of cloud computing and resource allocation techniques then completely different existing scheduling algorithms in cloud computing.

Keywords : Cloud Computing, Virtualization, Scheduling, Load Balancing, Resource Allocation, Resource Allocation Strategy, Ant Colony Optimization

I. INTRODUCTION

The goal of cloud computing is controlling, arranging, and reaching the hardware and software resources remotely. It recommends online data storage, infrastructure, and applications. It gives an independent platform as the software is not necessary to be setup on the PC. In cloud computing, virtualization is considered as the magic key; it represents a technology platform used for creating virtual instances through IT resources. A layer of virtualization software allows IT resources to provide multiple virtual images so that can be shared through multiple users [1]. As a result, cloud computing helps business applications to make portable and cooperative. National Institute of Standard and Technology (NIST) illustrates cloud computing with five characteristics, three service models and four deployment models as shown as figure 1.

In cloud computing, the **first layer** includes important five characteristics.There are on-demand self-service, broad network access, resource pooling,

rapid elasticity and measured service. **On-demand self-service** affords automatic computing capability to manage the systems, without any human interaction. **Broad network access** enables to access heterogeneous clients, such as laptops, smart phones, to connect to overall cloud systems through the network. **Resource pooling** is mainly available as pooling resources for connecting multiple consumers which can be able to dynamically assign and reassign according to consumer demand by using cloud systems. **Rapid elasticity** offers elastic and rapid provision of capabilities. It can quickly scale in and scale out automatically in order to support consumer's systems. **Measure service** the last characteristic comprises controlling, monitoring and reporting of resource usage [2].

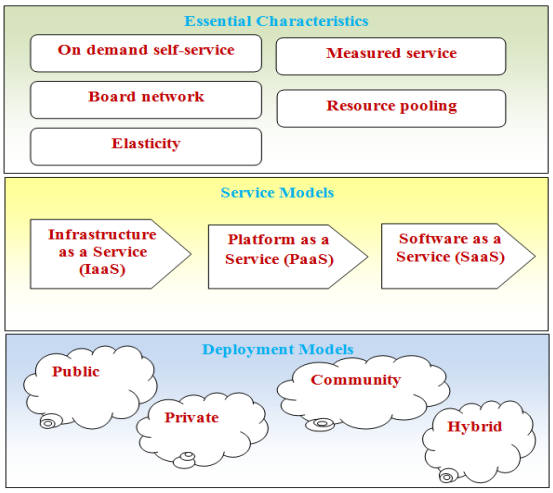

Figure 1. NIST visual model of cloud computing definition

The **second layer** of a cloud comprises cloud computing service models that are classified into three basic levels: 1.Infrastructure-as–a-Service (IaaS). 2. Platform-as-a-Service (PaaS). 3. Software-as-a-Service (SaaS). **Infrastructure-as Service (IaaS)** provides the capability to the consumer to provision processing, storage, networks, and other fundamental computing resources. A consumer is able to setup and run software, which can include operating systems through administrative access to manage applications and virtual machines but can't control the underlying cloud infrastructure [1]. **Platform as-a-Service (PaaS)** enables the lone developers to deploy web-based applications without buying actual servers and setting them up. **Software as-a-Service (SaaS)** enables a software distribution model to access a third party provider hosts applications and makes them available to customers over the internet. It can cut-down the total cost of software and hardware and development, maintenance, and operations. Now SaaS is provided by several organizations for example, Google, Microsoft, Amazon, Salesforce and Zoho [1].

Development model is defined as the communication between and cloud provider in public, private, hybrid and community cloud. In **public cloud,** the cloud infrastructure is available to manage large industry group or the general public and that is owned by an organization selling cloud services like Google, Microsoft and Amazon. There are many advantages of deploying, the public cloud model as cost effective, more reliable, highly flexible and scalable however low security problem is one of the main disadvantages of public cloud [3]. In **private cloud,** the cloud infrastructure is deployed to operate and maintain by using a specific organization. The operation may be in-house or with a third party access on the premises. The main benefits of the private cloud are as high privacy and security and full control but high cost and limited scalability are measured as a limitation. In **community cloud,** infrastructure is shared through a few associations and constructs a particular group. It can be controlled by an outside party or the associations [1]. In **hybrid cloud,** infrastructure consists of a number of clouds of different types; however, the clouds have the ability to allow data and/or applications to be transferred from one cloud to another through their interfaces. The hybrid cloud can be a combination of public and private clouds that support the requirement to maintain some data in an organization, and also provide few offer services in the cloud. (e.g., cloud blasting for load balancing between clouds) [3]

1.1 Virtualization

It is an important concept of cloud systems. Virtualization means "something which isn't real", however provides all the facilities of a real. It is the software implementation of a computer to execute different programs such as a real machine.Virtualization is opted to cloud, for the reason that using virtualization an end user may use different services of a cloud. The remote datacenter will offer different services in a full or partial virtualized manner. Virtualization is classified into two types that are found in case of clouds as given in [5]:

- Full virtualization
- Para virtualization

Full Virtualization: In this concept, an entire installation process of one machine is completed on another machine. It will lead to a virtual machine which is able to have all the software that is present within the actual server. Here the remote datacenter delivers the services in a fully virtualized manner. Full virtualization has been successful for several purposes as found out in [5]:

- Sharing a computer system among multiple users
- Isolating users from each other and from the control program
- Emulating hardware on another machine

Para virtualization: In Para virtualization, the hardware allows multiple operating systems to run on a single machine by efficient usage of system resources like processor and memory e.g. VMware software. At this point, all the services are not fully available, because rather than the services are provided partially.

System virtualization inserts a hardware abstraction layer on top of the hardware, which is named virtual or hypervisor machine monitor. Virtual Machines do not have permission to directly access the hardware. The hypervisor runs virtual machines in a non-privileged environment. Using system virtualization, multiple virtual machines, which can run various operating systems, can be run on a single physical machine. System virtualization is that the important technology that is used to offer IaaS, PaaS and SaaS resources.

A Virtual Machine Monitor (VMM), also known as Hypervisor, is software that securely protective during partitions the resources of a computer system into one or more virtual machines. A guest operating system is an operating system that runs under the control of a VMM rather than directly access on the hardware. The VMM runs in kernel mode, where a guest OS (operating system) runs in user mode. Different hypervisors support different features of the cloud. Hypervisors are available in many types:

- **Native hypervisors** that sit directly on the hardware platform are mainly used to gain better performance for individual users.
- **Embedded hypervisors** are integrated into a processor on a separate chip. This kind of hypervisor is however a service provider gains performance enhancements.
- **Hosted hypervisors** run as a distinct software layer above both the hardware and the operating system. Using this type of hypervisor is helpful both in public and private clouds to gain performance improvements [4].

II. RESOURCE ALLOCATION IN CLOUD COMPUTING

In this concept, resource allocation (RA) is a field taken into account in several computing areas like operating systems, datacenter management, and grid computing. RA deals with the division of available resources between the applications and cloud users in an economic and effective manner. It is one of the major challenging tasks in cloud computing supported the IaaS. Moreover, RA for IaaS in cloud computing provides many advantages: cost effective because users do not require to install and update

software or to access the applications, its flexibility allows access applications and data on any system within the world, and there are not any limitations of the medium or usage site. Additionally, there are two most important processes of RA via cloud computing.

- **Static Allocation:** Static Allocation schemes, assign fixed resources to the cloud user or application. The cloud user should know about the number of resource instances required for the application and what type of resources are requested and confirm the application's peak load requests. However the limitation for static allocation is generally affected by the under-utilization or over-utilization of computing resources supported to the normal workload of the application. This is not cost-effective and related to insufficient use of the resource during off-peak periods.
- **Dynamic Allocation:** In Dynamic Allocation schemes, offer cloud resources on the fly when application is requested, particularly to avoid over-utilization and under-utilization of resources. A possible disadvantage of resources are requested on the fly during that they could not be accessible. Therefore, the service supplier should allocate resources from different participating cloud data centers [6].

Resource allocation strategy (RAS) is related to combine cloud provider functions for utilizing and assigning limited resources within the boundaries of the cloud system in order to suit the require of the cloud application. The RAS should pass up the following situations as much as possible:

- **Resource contention:** This situation occurred at that time multiple users and applications attempt to allocate within the same resource simultaneous manner.
- **Resource fragmentation:** In this situation takes place applications cannot assign resources due to isolated resources being small items.
- **Scarcity:** This will occur at the same time multiple applications' requirements for the resources are high and limited resources, such as, I/O devices, requests for memory, CPUs, and the techniques for demand serve.
- **Over provisioning:** In this situation happens during the users and applications obtain more resources than the request to fit the quality of service (QoS) requirements.
- **Under provisioning:** This issues occurs when the users and applications obtain fewer resources than requested to fit the QoS requirements [6].

Cloud computing Systems increased as the number of users performs, the tasks to be scheduled in Cloud increased proportionally. As a result, there is a necessity for utilizing better algorithms to schedule tasks. Algorithms needed to schedule tasks are service oriented and modify in different environments. Task scheduling algorithms in cloud computing aim at minimum makespan of tasks with minimum resources efficiently. In Cloud computing utilizes low-power hosts to archieve high usability. A class of systems and applications are referred by the cloud computing that utilize distributed resources to execute a function in a decentralized manner. Cloud computing is to perform the computing resources (service nodes) on the network to speed up the execution process of difficult tasks that need to handle large-scale computation. Therefore, a task is to perform within the cloud computing should be considered as the selecting nodes for executing purpose [7]. A task is an active mode that utilizes a set of inputs to execute a set of outputs. In this Cloud computing process, user applications will perform after run on virtual systems where distributed resources are allocated dynamically. Dynamic load-balancing mechanism needs to allocate tasks to the processors dynamically as they arrive. Redistribution of tasks has occur perform when some processors are overloaded in any time. Each and every application is totally different in nature and independent where some application needs more CPU time to compute

complex task, and a few others may need additional memory to store data in an effective manner. Different scheduling algorithms can be used depending upon the kind of the task to be scheduled. The scheduling algorithms are mainly utilized for higher executing effectiveness and maintain the load balancing of the system.

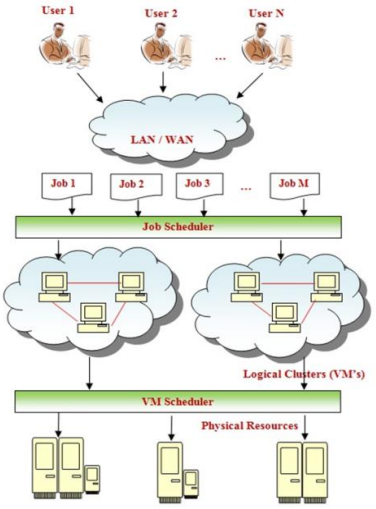

Figure 2. General View of Task Scheduling

2. 1 TASK SCHEDULING TYPES

- **Cloud Service Scheduling:** Cloud service scheduling is classified into two types. They are user level scheduling and system level scheduling [8]. In user level scheduling deals with issues raised through service provision between customers and providers. The system level scheduling handles resource management within the datacenter.
- **User Level Scheduling:** Market-based and auction-based schedulers are convenient for regulating the demand and supply of cloud resources. Market based resource allocation is more effective in cloud computing environment where resources are delivered and virtualized to user as a service. An AuctionNet for heterogeneous distributed environments is proposed from a suite of market oriented task scheduling algorithms [9].
- **Heuristic Scheduling:** Optimization issues are in class NP-hard. These issues can be solved through enumeration method, heuristic method or approximation method. In enumeration method, an optimal solution can be selected if all the possible solutions are enumerated and compared step by step. In case the number of instances is large, exhaustive enumeration is not possible for scheduling problems. In that case heuristic method could be a suboptimal algorithm to find reasonably good solutions in rapid manner. Approximation algorithms are used to find approximate solutions to optimized solution. These algorithms are mainly used for issues when occur on exact polynomial time algorithms are known.
- **Real Time Scheduling:** The primary objectives of real time scheduling are to raise throughput and reduce average response time rather than meeting deadlines. The real-time tasks are scheduled non-preemptively with the objective to increase the whole utility in [10]. Two different time utility functions (TUFs)-a profit TUF and a penalty TUF- are related with each task at an equivalent time. This way of approach is not only rewards for the early completions but also penalizes deadline misses or abortions of real-time tasks. In the same way of a preemptive algorithm is proposed in [11].
- **Workflow Scheduling:** A workflow enables the structuring of applications in a directed acyclic graph form [12], where each node performs the constituent task and edges represent inter task dependencies of the applications [13]. A single workflow usually consists of a set of tasks each of which may communicate with another task within the workflow. Workflow scheduling is one among the key issues within the management of workflow execution.

2.2 Existing Task Scheduling Algorithms

- **Opportunistic load balancing (OLB):** Without considering the job's execution time, it allocates a initial job to free machine. In this case more than one machine is getting free after that it assigns the job in arbitrary order to the processor. This scheduling mechanism runs in a faster manner.

The main advantage of this method is that it keeps the majority of machines get busy. However, it does not assure load balance.

- **Minimum Execution time (MET):** MET allocates each job to the machine that has the minimize expected execution time. It does not consider the availability of the machine and then the current load of the machine. The resources in Grid system have completely different computing power. All the smallest tasks is allocated to the same fastest resource redundantly creates an imbalance condition among machines. Therefore solution is considered in static manner.
- **Minimum Completion Time (MCT):** The algorithm calculates the execution time for a job on all machines by computing the machine's in convenient time and the expected completion time of the job on the machine. The job is selected on the machine with the minimum execution time. The MCT runs only one job at a time. This causes that particular machine may have the best expected execution time for any other job. The drawback of MCT will takes long time to calculate the completion time for a job.
- **Max-Min:** Max-min starts with a set of all unmapped tasks. Each machine is calculated with the execution time for each job. The minimum completion time for each job is selected through the machine. From the set, the algorithm maps the job with the overall maximum execution time to the machine. The above process is repeated until the remaining unmapped tasks were complete. Comparable to Min-min, Max min also considers all unmapped tasks at a time.
- **Min-Min:** Min-min algorithm begins with a set of all unmapped tasks. The completion time is calculated by each machine for each job. The minimum completion time for each job is selected by the machine. After that the job with the overall minimum completion time is selected and mapped to the machine. This process is repeated until the remaining unmapped tasks were complete. Compared to MCT, Min-min considers all unmapped tasks at a time. The drawback of Min-Min, too many jobs are assigned to a single node. This leads to response time and overloading issues of the job is not assured.

III. ANT COLONY OPTIMIZATION

Ant has the ability for searching an optimal path from nest to food[14,15,16].On the approach of ants moving, they lay some pheromone on the ground or any places; while an isolated ant encounter a previously laid trail, this ant can notice it and choose with high probability to follow it. For this reason, the trail is strengthened with its own secretion. The probability of ant chooses the simplest way is proportion to the concentration of a way's pheromone. Initialy to a way, the more ants choose, the way has denser pheromone, and finally the denser pheromone attracts more ants. By this positive feedback mechanism, ant can find an optimal way finally [17, 18, 19].

M. Dorigo, planned a scheme depend on ant Colony that is used for different optimization issues, wherever the ants of various colonies make solutions to a problem by sharing the quality information. ACO algorithm's working is comparable to the $64000 ants in which they try to find a shortest path between nest and food source. In ACO, multiple artificial ants make solutions to the optimization issue and share quality information, pheromone concentration on the trail traversed by ants. ACO acquires the real ant behavior as a basis. Therefore the deposition of the pheromone on the trail that ants traversed makes intelligence for others to be followed. More frequent traversed paths contain the higher pheromone concentration, whereas the less frequently used paths lose their importance because of less concentration of pheromone on that path. This depends on the assumption that the pheromone concentration fades away after a regular interval of

time. Thus the newly arriving ants intending to follow the paths with higher concentration take the advantage to traverse through the shortest path from source to destination. Ants move forward and backward following two ways [14]:

- ***Forward move*** – While moving forward, ants extract the food, or search food sources.
- ***Backward move*** – Whereas in backward move, ants pick up food from the food sources and span out back for food storage in the nest.

At time zero, ants are placed on different towns, the initial values τ_{ij} (0) for path intensity are set on edge (*i*, *j*). The primary part of each ant's tabu list is set to be equal to its starting town [20,21]. Afterward the *k*-ant moves from town i to town j with a probability that is defined as:

$$P_{ij}^{k}(t) = \begin{cases} \dfrac{[\tau_{ij}(t)]^{\alpha}[\eta_{ij}(t)]^{\beta}}{\sum_{k \in allowed_k}[\tau_{ik}(t)]^{\alpha}[\eta_{ik}(t)]^{\beta}} & if\ j \in allowed_k \\ 0 & otherwise \end{cases}$$

Where $allowed_k = \{N\text{-}tabu_k\}$, $tabu_k$ is the tabu list of *k*-th ant, $\tau_{ij}(t)$ is the pheromone value on edge (*i*, *j*), η_{ij} is the value of the heuristic value, and $\eta_{ij}(t) = 1/d_{ij}$. Where d_{ij} is the distance between node *i* and node *j*. α, β are two parameters that control the relative weight of the pheromone trail and heuristic value. As a final point the most optimal or effective path is selected and globally updated.

```
Procedure ACO
    begin
        Initialize the pheromone
    while (stopping criterion not satisfied) do
            Position each ant in a starting VM
        while (stopping when every ant has
            build a solution) do
        for each ant do
            Chose VM for next task by
            pheromone trail intensity
        end for
        end while
        Update the pheromone
    end while
end
```

Figure 3. Programming steps of the basic ACO

IV. CONCLUSION

In this cloud computing various kinds of resources are used, for example, Application Software, OS, Memory, CPU, etc. A cloud server, that has ample resources continually for its clients as resource pools, efficiently and dynamically allocates or de-allocates these resources, is considered smart work for its clients purpose. A number of algorithms are proposed within the past to solve the task-scheduling problem for heterogeneous network of computers. On the other hand, none of these algorithms can be extended to cloud computing systems and also heterogeneous computing systems. Since cloud computing systems have a high degree of unpredictability with respect to network bandwidth and resource availability, task scheduling algorithms for cloud computing systems must incorporate the latency caused by unpredictable resource availability. The current study involves surveying the different task scheduling algorithms developed for cloud environment.

V. REFERENCES

[1]. Bhaskar Prasad, Eunmi Choi and Ian Lumb, "*A Taxonomy and Survey of Cloud Computing Systems*", Fifth International Joint Conference on INC, IMS and IDC, 2009

[2]. Ikki Fujiwara," *Study on Combinatorial Auction Mechanism for Resource Allocation in Cloud Computing Environment*", Ph.D. thesis 2012.

[3]. Thomas Erl, Zaigham Mahmood, Ricardo Puttini, "*Understanding Cloud Computing*" in "*Cloud Computing Concepts, Technology and Architecture"* Second Edition September 2013, pp 27-49.

[4]. Miss. Rudra Koteswaramma, "Client-Side Load Balancing and Resource Monitoring in Cloud", International Journal of Engineering Research and Applications, November- December 2012

[5]. Ali M. Alakeel, A Guide to Dynamic Load Balancing in Distributed Computer Systems,

IJCSNS International Journal of Computer Science and Network Security, VOL.10 No.6, June 2010.

[6]. K Delhi Babu, D.Giridhar Kumar "Allocation Strategies Of Virtual Resources In Cloud Computing Networks" Journal Of Engineering Research And Applications,201,Pp.51-55.

[7]. Shu-Ching, Wang Kuo-Qin, Yan*(Corresponding author) Shun-Sheng, Wang Ching-Wei, Chen, ―A Three-Phases Scheduling in a Hierarchical Cloud Computing Network‖, 2011 Third International Conference on Communications and Mobile Computing, 978-0-7695-4357-4/11 © 2011 IEEE DOI 10.1109/CMC.2011.28

[8]. Fei Teng, "Resource allocation and scheduling models for cloud computing", Paris, 2011.

[9]. Han Zhao, Xiaolin Li, "AuctionNet: Market oriented task scheduling in heterogeneous distributed environments", IEEE, 2010

[10]. Shuo Liu, Gang Quan, Shangping Ren, "On-Line Scheduling of Real-Time Services for Cloud Computing", IEEE, 2010

[11]. Shuo Liu, Gang Quan, Shangping Ren, "On-line preemptive scheduling of real-time services with profit and penalty", IEEE, 2011

[12]. Zhifeng Yu and Weisong Shi, "A Planner-Guided Scheduling Strategy for Multiple Workflow Applications," icppw, pp.1-8, International Conference on Parallel Processing - Workshops, 2008.

[13]. J. Yu and R. Buyya, "Workflow Scheduling Algorithms for Grid Computing", Technical Report, GRIDS-TR-2007-10, Grid Computing and Distributed Systems Laboratory, The University of Melbourne, Australia, May 2007.

[14]. W. Ngenkaew, S. Ono and S. Nakayama, "Pheromone-Based Concept in Ant Clustering", Proc. 3rd IEEE Conf. on Intelligent System and Knowledge Engineering, 2008, 308-312.

[15]. Hui, Y., Xueqin, S., Xing, L.,Minghui, W,. "An improved ant algorithm for job scheduling in Grid " in Proceedings of the Fourth International Conference on Machine Learning and Cybernetics, DOI: 10.1109/ICMLC.2005.1527448 , pp. 2957-2961, 2005.

[16]. Manpreet Singh, "GRAAA: Grid Resource Allocation Based on Ant Algorithm" in 2010 Academy Publisher DOI: 10.4304/jait.1.3.133-135, 2010.

[17]. Ajay, K., Arnesh, S., Sanchit, A., and Satish, C., "An ACO Approach to Job Scheduling in Grid Environment" in Springer-Verlag Berlin Heidelberg 2010, SEMCCO 2010, LNCS 6466, DOI: 10.1007/978-3- 642-17563-3_35, pp. 286–295, 2010.

[18]. Li, L., Yi, Y., Lian, L., and Wanbin, S.,"Using Ant Colony Optimization for SuperScheduling in Computational Grid" in 2006 IEEE Asia-Pacific Conference on Service Computing, ISBN: 0- 7695-2751-5, 2006.

[19]. Liang, B., Yanli, H., Songyang, L., Weiming, Z., "Task Scheduling with Load Balancing using Multiple Ant Colonies Optimization in Grid Computing" in 2010 Sixth International Conference on Natural Computation (ICNC 2010), DIO: 10.1109/ICNC.2010.5582599, pp.2715-2719, 2010.

[20]. Bing, T., Yingying, Y., Quan, L., Zude, Z, "Research on the Application of Ant Colony Algorithm in Grid Resource Scheduling" in Wireless Communications, Networking and Mobile Computing, 2008. WiCOM '08. 4th International Conference, DOI: 10.1109/WiCom.2008.1354, pp.1-4, 2008.

[21]. Meihong, W., Wenhua, Z., "A comparison of four popular heuristics for task scheduling problem in computational grid" in Wireless Communications Networking and Mobile Computing (WiCOM), 2010 6th International Conference, DOI: 10.1109/WICOM.2010.5600872, 2010.

[22]. Ku Ruhana Ku-Mahamud, Husna Jamal Abdul Nasir, "Ant Colony Algorithm for Job Scheduling in Grid Computing" in ams, 2010 Fourth Asia International Conference on Mathematical/Analytical Modelling and Computer Simulation, pp.40-45, 2010